

Aterratge autònom d'aeronaus d'ala fixa basat en visió

Narcís Nogué Bonet

Resum— Per al meu treball de final de grau he decidit abordar el problema de l'atterratge autònom en aeronaus d'ala fixa sota normes de vol visual, ja que considero que els mètodes en ús avui en dia no es beneficien tan com podrien dels avenços més recents en tecnologia, principalment en els camps de la intel·ligència artificial i la visió per computador. Proposo que és factible guiar una aeronau d'ala fixa cap a un aterratge control·lat utilitzant només sensors instal·lats a bord i sense cap modificació necessària en la pista d'atterratge utilitzant la visió com a sensor principal, de forma que el procés queda molt simplificat i qualsevol pista és viable per a una aterratge autònom. Considero que en el transcurs del treball he demostrat i justificat que la meua proposta és viable tecnològicament i que solventa les limitacions presentades pels mètodes actuals. Durant el projecte també proposaré un mètode per a l'entrenament completament automàtic i no supervisat per a un sistema d'aquestes característiques.

Paraules clau— Aterratge, control autònom, aeronau, ala fixa, visió per computador, segmentació semàntica, aprenentatge no supervisat

Abstract— For my research work I have decided to address the problem of autonomous landing in fixed wing aircraft under Visual Flight Rules, as I consider that the methods in use today do not benefit as much as they could from the latest advances in technology, mainly in the fields of artificial intelligence and computer vision. I propose that it is feasible to guide a fixed-wing aircraft to a controlled landing using only on-board sensors and without any necessary runway modification using vision as the main sensor, greatly simplifying the process and making any runway available for autonomous landing. I consider that in the course of my research I have shown and justified that my proposal is technologically viable and that it solves the limitations presented by current methods. During my work I will also show a completely automatic and non-supervised learning method for a system such as the one I propose.

Keywords— Landing, autonomous control, aircraft, fixed-wing, computer vision, semantic segmentation, non-supervised learning



1 INTRODUCCIÓ - CONTEXT DEL TREBALL

COM indica el títol, el meu Treball de Final de Grau consisteix a crear un sistema de control autònom que sigui capaç d'aterrar un avió en una pista d'atterratge utilitzant únicament una càmera i altres sensors bàsics com acceleròmetres i giroscopis. Actualment la majoria de sistemes d'atterratge autònom necessiten modificacions substancials de la pista d'atterratge per instal·lar un sistema ILS (Instrument Landing System), dissenyat per permetre a una aereo-

nau aterrar de nit o en baixa visibilitat. Tot i això, hi ha un subgrup important dels aeroports que segueixen les normes VFR (Visual Flight Rules), on només es pot aterrar de dia i quan la visibilitat sigui suficient, ja que l'única informació que té el pilot és el contacte visual directe de la pista d'atterratge. La majoria d'aeroports petits, aeròdroms i pistes de muntanya cauen en aquesta categoria i per tant l'atterratge autònom per mètodes convencionals hi és de moment impossible. La solució que proposo deriva directament d'aquesta restricció: si la majoria de pistes d'atterratge estan pensades i dissenyades per a vol visual necessàriament un sistema d'atterratge autònom ha de ser capaç d'aterrar de forma purament visual per a poder-se considerar plenament autònom en tots els casos.

El principal benefici del sistema que proposo és la major flexibilitat per a aeronaus autònomes a l'hora d'escollir opcions d'atterratge, ja que no hi ha cap limitació que impe-

- E-mail de contacte: nnogue4@gmail.com
- Menció realitzada: Computació
- Treball tutoritzat per: Felipe Lumbreras Ruiz (Department of Computer Science)
- Curs 2020/21

deixi a un sistema com el que proposo aterrar en qualsevol pista on un humà seria capaç d'aterrar. Pistes petites i sense sistemes ILS instal·lats en aeròdroms o pistes de muntanya, o fins i tot pistes de terra que consisteixen d'esplanades allargades serien detectades i disponibles per a un aterratge autònom.

Una altra consideració que vull adreçar abans de començar és perquè he escollit treballar amb aeronaus d'ala fixa o "avions", quan un helicòpter o un multicopter (el que col·loquialment es coneix com a drone) poden enlairar i aterrar verticalment, presenten un control molt més simple, i per tant ja existeixen solucions comercials basades en visió per al control autònom d'aquest tipus d'aeronaus. El problema és que l'eficiència de les aeronaus d'ala giratòria és molt inferior a la de les aeronaus d'ala fixa i per tant estan menys capacitades per al vol de llarga distància, i per tant considero que una aeronau d'ala fixa és molt més adequada per a resoldre problemes del món real com el transport de mercaderies aeri autònom, per posar un exemple, i per això he decidit treballar sobre aquest tipus d'aeronau per al meu projecte.

2 OBJECTIUS

PER la naturalesa del projecte, els objectius del meu Treball de Final de Grau poden augmentar en complexitat molt ràpidament, i per tant els dividiré en dues seccions: els objectius necessaris per tenir un MVP (Minimum Viable Product), i la resta d'objectius opcionals per seguir expandint el projecte més enllà.

Objectius per a un MVP:

- Dissenyar i implementar un algoritme de control capaç d'aterrar un avió model si sap on és la pista d'aterratge.
- Dissenyar una simulació prou acurada d'un cas genèric d'aterratge, sobre la qual poder provar els algorismes de control i de detecció de la pista.
- Dissenyar i implementar una xarxa neuronal capaç de reconèixer qualsevol pista d'aterratge sobre la qual hagi estat entrenada directament.

Objectius addicionals:

- Construir un avió model capaç d'aterrar de forma autònoma a una pista d'aterratge.
- Dissenyar i implementar una xarxa neuronal capaç de reconèixer qualsevol pista d'aterratge que no hagi vist prèviament.

3 ESTAT DE L'ART

En la introducció ja he parlat una mica de com funciona l'aterratge autònom avui en dia, en aquesta secció entraré més en detall sobre els sistemes ILS i donaré una ullada a altres projectes similars al meu i com han resolt els problemes que se'm presenten. Molts dels sistemes que presentaré es comenten en aquest article (Gautam, Sujit, & Saripalli, 2014), així el que faré serà esmentar com funcionen i després comentar quines són les limitacions que considero que presenten que es podrien solventar amb un procés basat en visió per computador.

3.1 El sistema ILS

El sistema ILS (Sauta, Shatrakov, Shatrakov, & Zavalishin, 2019), anomenat Instrument Landing System o Sistema d'Aterratge Instrumental es considera un sistema d'ajuda per als pilots en situacions de baixa visibilitat, i només algunes categories d'ILS permeten aterratge automàtic a través d'un sistema Autoland. Els sistemes ILS es poden classificar en tres categories: CAT I, CAT II i CAT III, en funció de la precisió que proporcionen en el posicionament de l'aeronau, i només les categories II i III es consideren suficients per a aterratges automàtics.

Pel que fa al funcionament, un ILS consisteix en dos transmissors de ràdio situats a la pista d'aterratge. Un és el localitzador o *localizer* (LOC), que indiquen la direcció de la pista (en la figura 1 es mostren la pista i la senyal de ràdio vistes des de sobre).

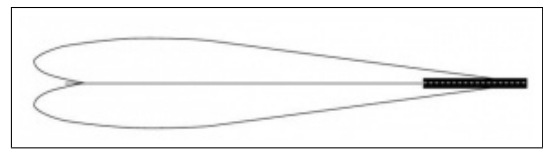


Fig. 1: Ràdio localitzador ILS

L'altra ràdio és la de pendent de descens o *Glide-Scope* (GS), que permet a l'aeronau controlar la ràtio de descens durant l'aproximació. (la figura 2 mostra la pista d'aterratge i la senyal de ràdio vistes de perfil).

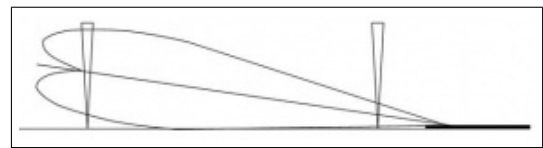


Fig. 2: Ràdio Glide-Scope ILS

El sistema ILS funciona bé i és molt robust, però necessita que les antenes de ràdio a la pista estiguin instal·lades i funcionin correctament, i avui en dia només els aeroports i aeròdroms amb més trànsit solen tenir aquest sistema, les pistes més petites i els aeròdroms en llocs remots solen quedar fora de l'equació pel que fa a aterratges amb ILS.

3.2 Sistemes basats en GPS

Una opció comuna és utilitzar un dispositiu GPS (Global Positioning System) a bord combinat amb un INS (Inertial Navigation System), per deduir la posició de l'aeronau respecte la pista d'aterratge. Normalment també es combina amb un baròmetre per determinar l'altitud ja que l'eix vertical és el menys precís per a un GPS. El principal problema que presenta aquest mètode és la falta de precisió i la falta de connexió entre el món real i les dades en que es basa el GPS de la aeronau. Tot i que és veritat que la precisió del sistema GPS ha augmentat a nivells suficients per resoldre el problema d'aterratge autònom, la connexió amb satèl·lits no sempre serà ideal i un error de precisió de pocs metres o menys d'un metre podria ser destructiu. Per altra banda, fins i tot assumint que un GPS tingués una precisió perfecta considero que no és suficient pel segon problema que he esmentat, la falta de connexió entre el món real i les

dades en que ens basem. Dit d'una altra manera, quan un avió es guia utilitzant un GPS persegueix un punt en l'espai en tres dimensions, i aquest punt necessàriament ha de ser pre-programat o descarregat d'una base de dades. Per tant, el punt en que l'avió creu que representa la pista i el punt on realment es troba la pista ara mateix en el precís moment d'aterrar no tenen per què ser el mateix, no hi ha res que garanteixi que seran el mateix més enllà d'un humà escrivint i actualitzant les dades cada cop que canvien o un sistema a la pista que en mesurés la posició GPS, i això trencaria la premisa d'aterratge autònom en pistes no modificades.

I evidentment una pista no canvia de posició regularment, per això aquest mètode és viable, però pot passar. Hi pot haver error humà en entrar les dades, una pista es pot reconstruir per mantenir-la en bon estat i la seva posició pot canviar lleugerament, poden aparèixer obstacles en la trajectòria d'aterratge, i en cap d'aquests casos l'aeronau seria capaç d'aterrar de la forma prevista.

Considero llavors, que el meu mètode solventa aquest problema perquè extreu les dades directament del món real en el precís moment de l'aterratge, i per tant elimina completament la capa de les dades emmagatzemades del problema i assegura que les dades sobre les que treballa es corresponen directament amb el món real.

3.3 Sistemes basats en visió

En aquesta secció parlaré de sistemes que s'han proposat anteriorment d'aterratge autònom basats en visió, molt més similars al que jo proposo. La varietat de sistemes en aquest sub-grup és gran i va desde sistemes que requereixen modificacions visuals de la pista fins a vol completament autònom basat únicament en visió

Una alternativa és la d'utilitzar emissors de llum infraroja des de la pista (Wenzel, Rosset, & Zell, 2010) o plataforma d'aterratge com es proposa en aquest article on es demostra que aquest mètode és viable per a guiar multicòpters cap a un objectiu i aterrar en vertical. A més també es pot veure en aquest article que aquest mètode es pot implementar fàcilment amb una mentalitat low-cost i amb electrònica àmpliament disponible, de la mateixa forma que pretenc fer jo per a una demostració del sistema en el món real. Tot i així no es parla de l'aterratge d'aeronaus d'ala fixa en aquest cas.

En aquests altres exemples (Saripalli, Montgomery, & Sukhatme, 2003) i (Saripalli & Sukhatme, 2003) es pot veure com un sistema basat únicament en visió i sense modificació de la pista és viable per al control i aterratge d'un helicòpter amb bons resultats, fins i tot en el cas d'aterrar en una pista mòbil on una posició GPS seria de poca utilitat. I en aquest exemple (Lange, Sunderhauf, & Protzel, 2009) es pot veure com aterrar utilitzant visió per computador és viable per a multicòpters en entorns on ni tan sols hi ha senyal GPS.

En aquest cas (Kim et al., 2013) podem veure un exemple implementat a bord d'aeronaus d'ala fixa, més semblant al que proposo, però on l'aterratge final es fa impactant contra una xarxa a terra i no en una pista d'aterratge, i per tant és un mètode viable per a petites naus no tripulades i en localitzacions on s'hagi instal·lat la xarxa d'aterratge, però no per a aeronaus a gran escala i en aterratges en pistes habituals i no modificades per a aquest cas en concret.

Un altre exemple treballa amb aeronaus d'ala fixa (Kong et al., 2014) proposa un sistema on els elements de visió consisteixen en càmeres instal·lades a la pista d'aterratge que permeten deduir la posició, rotació i velocitat de les aeronaus que s'aproximadament i guiar-les per a l'aterratge. El sistema funciona bé, però requereix modificacions en la pista i això és precisament el que intento evitar.

Finalment, en l'últim exemple que vull comentar (Le Bras, Hamel, Mahony, Barat, & Thadasack, 2014) es proposa un sistema d'aterratge completament autònom, per a aeronaus d'ala fixa i amb sensors visuals a bord de l'aeronau i sense modificacions necessàries a la pista d'aterratge, però només implementa les maniobres d'aproximació a la pista sense implementar les maniobres de flare i touchdown, i per tant es queda curt de ser un aterratge autònom complet.

Per tant, es pot veure primer de tot que en el camp dels aterratges autònoms basats en visió per computador hi ha una clara diferència entre les aeronaus d'ala fixa i les aeronaus d'ala giratòria, ja que les d'ala giratòria presenten molta més llibertat de control i maniobrabilitat, es poden moure més lentament i poden parar-se en l'aire. Per tant és significativament més fàcil aterrar de forma autònoma i utilitzant la visió en una aeronau d'ala giratòria i per tant hi ha molta més recerca centrada en aquest cas d'ús. Per altra banda en l'aterratge autònom d'ala fixa es troba molta menys recerca i en la majoria de casos encara es requereix una modificació de la pista d'aterratge o no s'implementa l'aterratge complet en una pista, ja que les maniobres finals són les més difícils i delicades.

4 METODOLOGIA

En aquesta secció presentaré el procés que he seguit durant la meua recerca per a resoldre el problema que m'havia proposat. Hi ha quatre tasques principals que he abordat: la obtenció i generació automàtica de dades per a l'entrenament, l'entrenament d'una xarxa neuronal sobre les dades, la creació d'un simulador prou similar al món real per provar-hi el sistema complet, i la implementació d'un algoritme de control capaç de guiar la aeronau utilitzant només la sortida de la xarxa neuronal i un giroscopi.

4.1 Generació automàtica de dades per a aprenentatge no supervisat

He montat un sistema que genera homografies d'imatges satèl·lit per generar una sensació de perspectiva que considero que serà suficient per entrenar un model, i només necessita les coordenades de les quatre cantonades de la pista d'aterratge per fer-ho (de l'estil de les imatges en la figura 3). I com que conec les coordenades de la pista d'aterratge i les seves cantonades puc generar la màscara automàticament també.

Un cop tinc la imatge del terreny generada he afegit un procés que distorsiona la imatge només a la part que correspon al terra, amb la intenció d'evitar que una xarxa neuronal pogués aprendre a reconèixer les homografies de les pistes utilitzades en el dataset, i d'aquesta manera evitar l'overfitting de la xarxa. Afegir aquest pas ha donat molt bons resultats i les xarxes que he pogut entrenar generalitzen molt millor quan es troben amb imatges del món real.

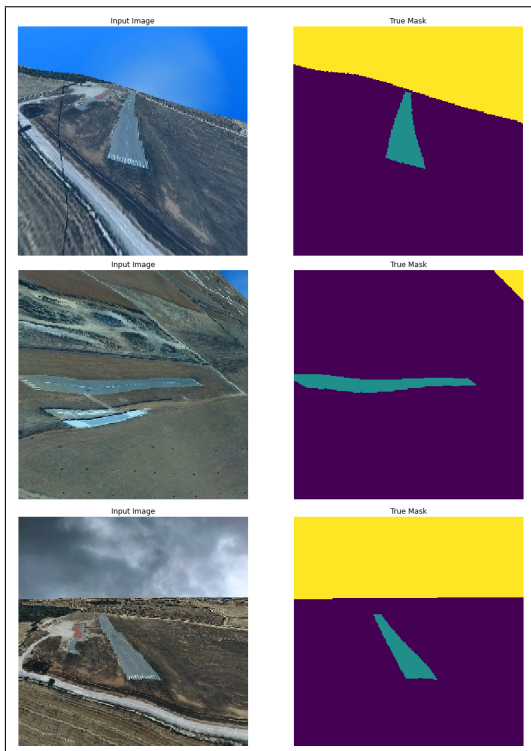


Fig. 3: Exemples d'imatges generades automàticament i les prediccions esperades

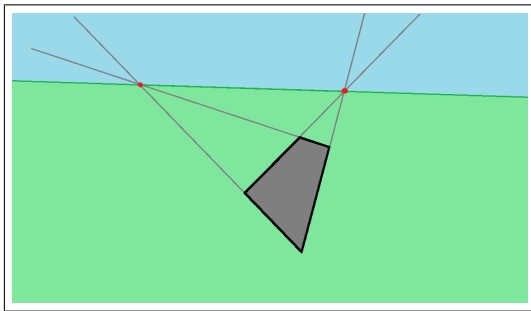


Fig. 4: Exemple de detecció automàtica de l'horitzó en una homografia

Tot i així, només puc generar una imatge cada 2 segons més o menys, ja que cada imatge fa unes 100 crides asíncrones al servei d'imatges satèl·lit i estic limitat pel major temps de resposta, per tant també he implementat un sistema d'augmentació de dades que fa zoom en diferents posicions de cada imatge i extreu unes 120 imatges noves per a cada imatge original.

Quan genero les homografies també es genera un horitzó ja que hi ha un punt on els detalls de la imatge es fan infinitament petits, i si busqués imatges més enllà d'aquest punt els resultats es distorsionen i deixen de tenir sentit, perquè no hi ha res més enllà de l'horitzó i no té sentit intentar generar-ho. Per tant és necessari detectar la línia de l'horitzó automàticament per a cada imatge que genero. Per fer-ho utilitzo les cantonades de la pista en la posició que queden després de fer la homografia. Si genero una recta que passi per cada parell de cantonades i les allargo fins a l'infinit per trobar els punts on es creuen. Hi ha quatre rectes que generen dos punts d'intersecció. La recta que uneix els dos punts d'intersecció representa l'horitzó tal i com es

mostra en la figura 4. Un cop detectada la recta que representa l'horitzó em cal saber si la pista es troba per sota o per sobre d'aquesta recta (es generen homografies en que la imatge està invertida). Per saber-ho només cal comprovar on es troba una de les cantonades de les quals coneixem la posició en la imatge final. Per evitar buscar infinites imatges just a l'horitzó també desplaço la recta de l'horitzó cap a la pista en un 5% del tamany de la imatge, és a dir, no genero el terreny fins a arribar a l'horitzó real ja que el detall es fa infinitament petit.

Cal destacar que en el gràfic d'exemple (4) la pista està molt distorsionada per tal que els punts d'intersecció quedin dins la imatge. En realitat en la majoria de casos els punts d'intersecció queden a milers o desenes de milers d'unitats (píxels, si té sentit parlar de píxels més enllà de la imatge) fora la imatge, però el resultat és el mateix i la línia d'horitzó es detecta correctament igualment.

Finalment, un cop determinada la recta final de l'horitzó ompló el cel amb un fragment aleatori d'una imatge d'un cel aleatori d'entre una col·lecció d'imatges de cels que he descarregat, i modifico el color del terra en funció del color del cel per simular la il·luminació que generaria el cel en concret (llum grisa per cel nuvol, llum blava per cel blau, etc). Cal destacar que aquest últim pas no només serveix per a generar una imatge més similar a la realitat, sinó també per afegir més variabilitat a les homografies del terra i evitar l'overfitting.

Després de tot aquest procés, generar la màscara desitjada és trivial ja que tinc les quatre cantonades de la pista en la imatge final i la equació de la recta de l'horitzó, per tant només cal pintar una màscara unilitzant aquesta informació i distorsionar-la de la mateixa forma en que s'ha distorsionat la homografia perquè tot encaixi correctament. El resultat final es pot veure en la figura 3.

4.2 Entrenament d'una xarxa neuronal

Al principi del projecte vaig plantejar varis models d'aprenentatge per a la detecció de la pista, i sobretot en vaig discurrir més en profunditat tres:

- Detectores de característiques de l'estil de Sift o Surf (Khan, McCane, & Wyvill, 2011), són molt ràpids però tenen dificultats detectant característiques complexes en totes les condicions possibles.
- Detectores de segmentació semàntica, principalment U-Net (Ronneberger, Fischer, & Brox, 2015), pot ser més lent però és flexible i pot donar prediccions fiables en entorns molt complexos.
- Entrenament End-to-End (Bojarski et al., 2016), on la intenció és que una sola xarxa neuronal resolgui el problema de visió i el de control alhora. Pot donar molt bons resultats però és més difícil d'entrenar correctament.

Finalment vaig continuar el projecte amb un detector de segmentació semàntica, que consisteix en una xarxa neuronal que rep com a input una imatge i retorna una màscara que separa les parts de la imatge que són pista d'aterratge i les que no ho són (figura 5).

El model de xarxa neuronal que he utilitzat és el model pix2pix de TensorFlow, que és una arquitectura basada en

l'arquitectura U-Net (Ronneberger et al., 2015) i que permet fer segmentació semàntica a un cost computacional relativament reduït.

Tot l'entrenament l'he fet en Google Colab perquè m'ha donat millors resultats, així que el codi a Github no està sempre actualitzat. Es pot veure el codi més recent en el que estic treballant [aquí](#).

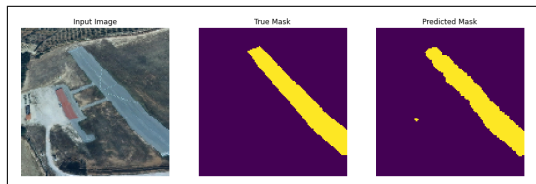


Fig. 5: Exemple d'una imatge, la predicció esperada i la predicció real

Tot el codi d'entrenament està integrat amb el codi de generació d'homografies de manera que un cop activat, el programa genera homografies aleatòries i úniques i després entrena un model nou sobre aquestes homografies o entrena un model ja entrenat sobre el nou set d'imatges, i per tant tot el procés d'entrenament és automàtic i no requereix input o supervisió humana per funcionar, i per tant es pot estalviar completament l'aspecte més costós de la intel·ligència artificial que és la obtenció i anotació manual de les dades.

4.3 Creació d'un simulador

Per a aquesta tasca vaig decidir utilitzar el motor de creació de videojocs Unity per a fer la simulació d'un avió, un terreny i una pista d'aterratge (figura 6). També vaig implementar un motor físic per a simular el moviment de l'avió segons la posició de les seves diverses superfícies de control.

Per simular les forces que actuen sobre l'avió vaig utilitzar equacions simplificades per al fregament amb l'aire i la força d'elevament que causa cada una de les superfícies de control de l'avió en funció de la velocitat i l'angle d'atac.

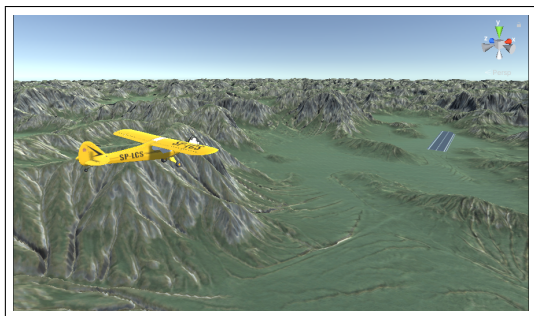


Fig. 6: Simulació de l'avió i la pista d'aterratge

Per a provar el meu sistema de control autònom sobre el simulador necessitava una forma de controlar Unity amb Python, i com que Unity no suporta Python directament he montat un programa de control en Python que es comunica amb el controlador de l'avió a Unity per Sockets. La idea és que pugui rebre la imatge que captura la càmera de l'avió simulat (figura 7), processar-la i enviar les posicions de les superfícies de control que considera necessàries.



Fig. 7: Simulador a l'esquerra i la imatge rebuda pel controlador a la dreta

4.4 Tasca 5: Integració del model i la simulació i programar el control de l'avió

Un cop implementat el simulador i una xarxa neuronal que generalitzi bé vaig dedicar-me a integrar-los tots dos, de manera que el servidor en python pugués veure l'estat del simulador, analitzar-lo utilitzant els models entrenats prèviament i diferents tècniques d'anàlisi d'imatges, i enviar una resposta al simulador per poder controlar l'avió, tot tant en temps real com fos possible. Desde instal·lar i provar diferents configuracions de drivers de Nvidia i versions de les diferents llibreries que utilitzo fins a utilitzar multithreading per separar la comunicació amb el simulador i l'anàlisi pròpiament dit, finalment tot el bucle funciona 10 vegades cada segon, que és més que suficient per el que necessito.

Per analitzar una imatge el primer que faig és fer una predicció de segmentació semàntica amb la xarxa neuronal per obtenir una aproximació del que és la pista i el cel, després extrec només una màscara per la pista i la erosiono i la dilato seqüencialment per eliminar el soroll. Com que normalment no s'elimina tot el soroll assumeixo que el bloc més gran és la pista i en busco el punt mig i els punts màxims i mínims per trobar les cantonades. Finalment busco el punt mig de l'aresta de la pista més propera a l'avió, ja que aquest punt és el que servirà de referència per al controlador a l'hora de dirigir l'avió.

Per la banda del controlador, llavors, cada dècima de segon arriba un punt representat per dues coordenades, i que representa el centre de l'aresta més pròxima de la pista en el pla en dues dimensions que veu la càmera frontal de l'avió. Com que conec la posició de la càmera i l'angle actual de l'avió (recordo que l'avió incorpora un giroscopi), puc traduir aquestes coordenades en un vector que indica la direcció i el sentit cap a la pista desde l'avió. Aquest vector és el que es seguirà fins que arribi el següent frame i es repeteixi el procés. Per eliminar soroll, cal remarcar que en realitat el vector es calcula amb la mitjana dels deu últims punts que han arribat, ja que com que el moviment de l'avió no és brusc un delay de aproximadament un segon no impossibilita un bon resultat i ajuda molt en reduir moviments bruscs per punts que no han estat ben detectats en algun frame.

Per assegurar que l'avió es dirigeix en la direcció correcta sense fer moviments bruscs i sense oscil·lacions massa grans utilitzo quatre controls PID (Proporcional, Integral i Derivatiu), dos per cada eix. El primer calcula la inclinació desitjada de l'avió en funció de la posició objectiu i la posició actual, i el segon calcula la posició de les superfícies de control per aconseguir la inclinació desitjada, tant per l'eix de gir dreta-esquerra (roll) com per l'eix amunt-avall

(pitch).

En la figura 8 es veu tot el procés, amb el simulador a la dreta i la imatge que rep, la màscara de la pista analitzada i la màscara de la pista i el cel sense processar a l'esquerra. En la imatge del centre a la columna de l'esquerra es veuen diversos punts que representen la pista, el més important és el punt central dels tres inferiors, que representa l'inici de la pista i és el punt cap on l'avió intenta dirigir-se.

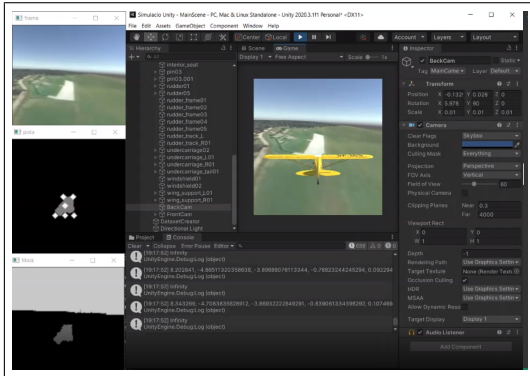


Fig. 8: Simulador a la dreta i procés d'anàlisi de la imatge a l'esquerra

Finalment, just abans d'aterrar l'avió perd de vista la pista perquè la pista ocupa la gran majoria de la imatge com es pot veure en la figura 9. Per tant mitjançant un sensor ultrasònic de distància amb un rang de fins a quatre metres detecto quan l'avió està just a sobre el terra i ordeno al controlador que mantingui l'avió anivellat i disminuint l'alçada a poc a poc utilitzant la potència del motor, d'aquesta manera, i assumint que l'avió estava completament encarat a la pista quan s'ha detectat el terra, aconsegueixo un aterratge suau.

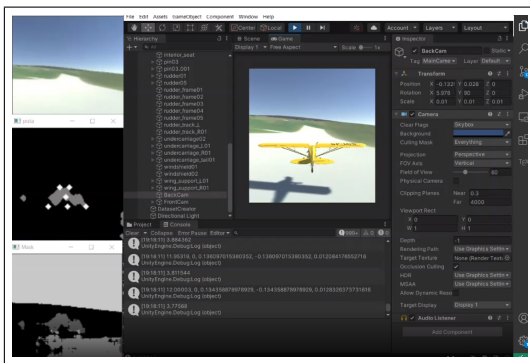


Fig. 9: Simulador en els moments finals just abans d'aterrar

4.5 Comentari adicional sobre el desenvolupament

Finalment vull comentar que com es pot veure hi ha un objectiu dels dos objectius opcionals que no he pogut dur a terme per falta de temps, el de construir un avió model que integrés tot el meu projecte i el posés a prova en el món real i en l'entorn extremadament inestable d'un avió a petita escala. Aquest objectiu era evidentment el més ambiciós però tenia certa importància per mi perquè en la majoria de projectes que he dut a terme mai he vist els resultats sortir de l'entorn controlat d'una pantalla d'ordinador,

i aquest projecte en concret era ideal per dur-lo al món real i veure resultats d'una forma encara més satisfactòria. Per això vull seguir amb aquest projecte més enllà d'aquest curs acadèmic i espero aquest estiu poder continuar amb la construcció d'un avió que en algun moment podrà aterrar de forma totalment autònoma utilitzant tan sols una càmera. Per ara el que he pogut desenvolupar és la plataforma que integra tota la electrònica principal de l'avió i 'només' faltaria construir-hi un avió al voltant i adaptar tot el codi perquè funcioni en el nou entorn. I després a continuar desenvolupant perquè realment tinc moltes idees per seguir expandint aquest projecte, però d'això en parlaré més a les conclusions.

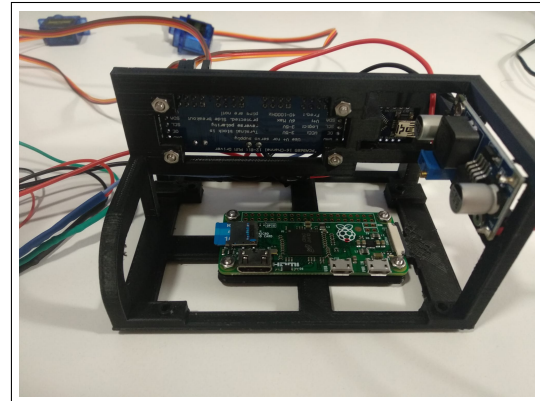


Fig. 10: Plataforma que integra una Raspberry Pi Zero, un Arduino nano i un controlador pwm per servos, amb suport per a una Jetson Nano

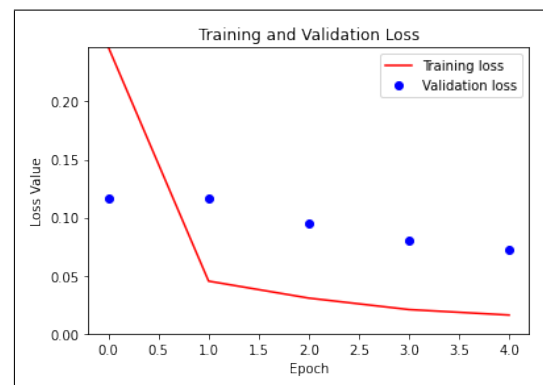


Fig. 11: Mètrica Loss en el procés d'entrenament, he trobat que entre 5 i 6 iteracions són suficients per a entrenar un bon model sense overfitting.

5 RESULTATS

Per valorar els resultats del projecte em basaré directament en els objectius que m'havia proposat i comentaré un per un aquells que en aquesta fase del projecte mostren progrés o estan completats.

Parlaré dels objectius referents al Minimum Viable Product, o sigui aquells necessaris per a considerar el projecte completat, ja que són aquells que finalment he pogut desenvolupar plenament, i els comentaré en l'ordre aproximat en que he abordat i resolt els reptes que proposaven:

- El primer objectiu proposa dissenyar una xarxa neuronal capaç de reconèixer aquelles pistes sobre les que ha estat entrenada. He aconseguit entrenar el model pix2pix, que deriva d'una xarxa UNet fins a aconseguir una accuracy en la validació del 98% sense overfitting, i he demostrat que les prediccions són prou precises i consistents com per a permetre una aproximació a la pista i una aterratge autònoms. En la figura 5 es pot veure un exemple de les prediccions que obtinc actualment, i en la figura 11 es pot veure el resultat del procés d'entrenament.
- El segon objectiu consisteix en dissenyar un simulador prou exacte com per permetre provar una integració completa del sistema. Aquest objectiu també el considero completat en aquest punt del projecte, com he comentat en el punt 3.3 he dissenyat la simulació en Unity i estic satisfet amb el resultat ja que sense assumir gairebé res sobre com hauria de funcionar un avió i simplement implementant la simulació de les físiques involucrades en el problema he aconseguit un simulador que es comporta de forma prou similar a un avió i que implementa totes les mecàniques rellevants al problema que vull resoldre.
- Finalment, el tercer objectiu per a un MVP consistia en dissenyar un algoritme de control capaç de controlar l'avió amb la informació que rep de la xarxa neuronal i de sensors simples. Aquest objectiu també està completat, de manera que utilitzant tan sols una càmera, un giroscopi i un sensor ultrasònic que pot detectar el terra desde fins a quatre metres d'alçada l'avió es pot guiar de forma autònoma fins a realitzar un aterratge controlat en la pista d'aterratge, tal i com es pot veure en aquest [vídeo](#) de la pàgina de GitHub del projecte.

6 CONCLUSIONS

Vull començar dient que considero que a l'hora de triar el projecte que volia abordar pel meu TFG vaig ser conscientment ambiciós, tot i que sabia que probablement no seria a temps d'acabar tots els objectius que em proposava al nivell de qualitat que desitjava, i efectivament he hagut d'agafar draceres i assumir l'actitud de optimitzar tot el projecte al voltant de tenir una demostració de viabilitat el més ràpid possible perquè sinó hagués estat impossible per a mi complir els objectius a temps.

Dit això, estic increïblement satisfet amb els resultats que he obtingut fins al moment, tots els passos que em vaig proposar a l'inici del projecte han encaixat perfectament entre ells i he pogut desenvolupar-los exactament en el temps previst, i he demostrat que un avió pot aterrar seguint el mètode que vaig proposar basant-se principalment en la visió.

Vull aprofitar per recordar quin és el motiu pel que considero que és necessari que un avió pugui aterrar només amb visió i mesures internes com podrien ser giroscopis o sensors de proximitat: actualment tots els mètodes d'aterratge autònom es basen en infraestructura instal·lada a la pista d'aterratge o bé en les coordenades GPS de la pista. En el primer cas el problema és obvi, no totes les pistes tenen la infraestructura necessària per a un aterratge autònom i això limita els aeroports on aquest mètode és viable. En el segon

cas considero que el problema és que es confia en dades externes que no tenen necessàriament una correlació directa amb el món real, les coordenades podrien ser errònies o la pista podria haver canviat lleugerament i l'avió no té forma de contrastar i comprovar les dades GPS que rep.

Per tant considero que una aproximació visual és l'únic mètode que permet que l'avió extregui tota la informació que necessita directament del món que l'envolta, i això assegura que la informació sobre la que treballa representa exactament el món en el que es mou. Per tant aquest mètode permet aterrar en qualsevol aeroport sota les Regles de Vol Visual, i elimina les limitacions que presenten els mètodes més utilitzats actualment.

Finalment, m'agradaria destacar el potencial que crec que un aterratge visual té, la demostració que presento actualment és extremadament limitada i no representa el total de possibilitats que s'obren quan un vehicle autònom es pot guiar per la visió. En un futur a mesura que continuï desenvolupant aquest projecte m'agradaria arribar al punt en que l'avió pogués fer un model en tres dimensions del món que l'envolta utilitzant la informació que rep desde la càmera, i preparar i seguir una trajectòria en aquest model 3D, de manera que pogués seguir la ruta més òptima o patrons d'aproximació preestablerts, i considero que tot això és possible utilitzant únicament la visió.

AGRAÏMENTS

Hi ha varies persones que m'han ajudat molt durant tot aquest projecte, ja sigui guiant-me en els aspectes tècnics del projecte, donant-me facilitats perquè pogués fer tota la feina sense impediments, o donant-me suport i animant-me sempre que he sentit que em quedava enrere, i vull agrair-los enormement tot el que han fet per mi durant tot aquest temps:

- Al meu tutor en aquest TFG, el Felipe Lumbreras, pel seguiment constant que ha fet del meu projecte, amb una reunió cada dijous sens falta des de l'inici, per haver-me guiat durant el desenvolupament en tots els àmbits on tenia dubtes sobre com continuar, i per haver-se interessat sincerament en el meu projecte des que el vaig proposar i totes les idees genials que m'ha proposat sobre com continuar-lo que m'encantaria dur a terme en un futur proper.
- Al Ramón, el meu encarregat al meu lloc de treball durant tot el curs, per haver-se interessat pel meu projecte i haver-me donat tota la flexibilitat que ha pogut durant el transcurs del meu TFG quan em faltaven hores per arribar a una entrega.
- A la meua parella, per haver estat el pilar principal de suport des del primer dia, per haver-me animat sempre que he sentit que no arribava a alguna entrega, per haver-me ofert ajuda sempre que la he necessitat i per haver-me comprès i esperat pacientment en les setmanes en que gairebé no podem ni veure'ns abans d'alguna entrega.
- A la meua família, per tot l'interès que han mostrat en el meu projecte i l'entusiasme i suport incondicional que m'han donat des del primer dia.

REFERÈNCIES

- Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., ... others (2016). End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.
- Gautam, A., Sujit, P., & Saripalli, S. (2014). A survey of autonomous landing techniques for uavs. In *2014 international conference on unmanned aircraft systems (icuas)* (pp. 1210–1218).
- Khan, N. Y., McCane, B., & Wyvill, G. (2011). Sift and surf performance evaluation against various image deformations on benchmark dataset. In *2011 international conference on digital image computing: Techniques and applications* (pp. 501–506).
- Kim, H. J., Kim, M., Lim, H., Park, C., Yoon, S., Lee, D., ... Kim, Y. (2013). Fully autonomous vision-based net-recovery landing system for a fixed-wing uav. *IEEE/ASME Transactions On Mechatronics*, 18(4), 1320–1333.
- Kong, W., Zhou, D., Zhang, Y., Zhang, D., Wang, X., Zhao, B., ... Zhang, J. (2014). A ground-based optical system for autonomous landing of a fixed wing uav. In *2014 IEEE/RSJ international conference on intelligent robots and systems* (pp. 4797–4804).
- Lange, S., Sunderhauf, N., & Protzel, P. (2009). A vision based onboard approach for landing and position control of an autonomous multirotor uav in gps-denied environments. In *2009 international conference on advanced robotics* (pp. 1–6).
- Le Bras, F., Hamel, T., Mahony, R., Barat, C., & Thadasack, J. (2014). Approach maneuvers for autonomous landing using visual servo control. *IEEE Transactions on Aerospace and Electronic Systems*, 50(2), 1051–1065.
- Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 234–241).
- Saripalli, S., Montgomery, J. F., & Sukhatme, G. S. (2003). Visually guided landing of an unmanned aerial vehicle. *IEEE transactions on robotics and automation*, 19(3), 371–380.
- Saripalli, S., & Sukhatme, G. S. (2003). Landing on a moving target using an autonomous helicopter. In *Field and service robotics* (pp. 277–286).
- Sauta, O., Shatrakov, A., Shatrakov, Y., & Zavalishin, O. (2019). Instrumental landing systems. In *Principles of radio navigation for ground and ship-based aircrafts* (pp. 65–71). Springer.
- Wenzel, K. E., Rosset, P., & Zell, A. (2010). Low-cost visual tracking of a landing place and hovering flight control with a microcontroller. *Journal of Intelligent and Robotic Systems*, 57(1), 297–311.